

University of Bahrain  
Department of Computer Science

College of Information Technology  
ITCS332: Concepts of Programming Languages

**QUIZZ#1: Chapter 1**

\*\*\*\*\*

**True/False Questions:**

1. \_\_\_\_\_ Java programs are less reliable than semantically equivalent C++ programs.
2. \_\_\_\_\_ The larger the number of operators in a language, the worse the writeability and the better the readability of programs.
3. \_\_\_\_\_ Demanding all subscripts of array references to be checked against acceptable range increases reliability and decreases the cost of a PL.
4. \_\_\_\_\_ Run-time type checking is cheaper and desirable than compile-time type checking.
5. \_\_\_\_\_ APL designers traded readability for writeability.

**Fill in blanks Questions:**

6. Name a programming language that fits business domain is \_\_\_\_\_.
7. A program is said to be \_\_\_\_\_ if it conforms to its specifications under all circumstances.
8. The main bottleneck in von Neumann computers is \_\_\_\_\_.
9. Type checking may be performed during \_\_\_\_\_ or \_\_\_\_\_.
10. The presence of two or more distinct referencing methods or names for the same memory location is called \_\_\_\_\_.
11. An example of process abstraction is \_\_\_\_\_.

## QUIZZ#1: Chapter 1

\*\*\*\*\*

## Fill in blanks Questions:

1. The language design is influenced by 2 factors: \_\_\_\_\_  
and \_\_\_\_\_.
2. The bottleneck of interpreting systems is repeating code executing.
3. The compiler phase that converts the source program characters into lexical units is called Implementation.
4. The language category that mainly based on the concepts of data abstraction and inheritance is called object oriented ✓.
5. Name a programming language that fits the scientific application is ALGOL.
6. A JAVA is a program that processes a program immediately before it is compiled.

## True/False Questions:

1. T ✓ JAVA designers traded pointers for language reliability.
2. T ✓ Array subscripts are usually checked against the acceptable ranges during the link-time.
3. T ✓ Programs executed in compiled systems are faster than those in interpreting ones.
4. T ✓ Nowadays, the most commonly used programming languages are compiled languages.
5. F ✓ Demanding all type checking to be performed increases both the reliability and the cost of a programming language.

Student Name:

Student id:

Section #:

University of Bahrain  
Department of Computer Science

College of Information Technology  
ITCS332: Concepts of Programming Languages

**QUIZZ#5: Chapter 5\_Names**

\*\*\*\*\*

**QUESTION ONE: Fill in blanks**

- 13) The process of automatic conversion of a variable type to another type by a compiler is called coersion.
- 14) The binding of a variable to a value at the time it is bound to storage is called Static binding ~~Initialization~~ dynamic scoping.
- 15) In a Heap ~~dynamic~~ language, the referencing environment is the local variables plus all visible variables in all active subprograms.
- 16) The type of a variable defines 2 things: the range of value and operation defines on that var context of the reference.
- 17) The type of an object can be determined using 3 ways: automatic like fortran when assigned to assignment statement, and declaration statement. (b, k, L, m otherwise need)
- 18) "int" type in C++ is bound to a range of values at language implementation time.
- 19) The explicit heap ~~dynamic~~ Alias variable has 2 variables associated with it.
- 20) dynamic ~~dynamic~~ variables are bound to storage when their declaration statements are elaborated. Stack dynamic

**QUESTION TWO: TRUE / FALSE**

- 21) F A program that heavily uses aliases is more readable than a one without aliases.
- 22) F A procedure P1 is a static parent of a procedure P2 if P1 contains the definition of P2.
- 23) F A call to a function may be bound to the function code at link time.
- 24) F Subrange variables of integer type are compatible with integer types. not

Student Name:

Student id:

Section #:

University of Bahrain  
Department of Computer Science

College of Information Technology  
ITCS332: Concepts of Programming Languages

### QUIZZ#5: Chapter 5\_Names

#### QUESTION ONE: Fill in blanks

- 1) The 2 types of type compatibility are: name compatibility and structure compatibility.
- 2) The lifetime of a dynamic variable is while the subprogram is active.
- 3) Dynamic Scope is based on the calling sequence of program units, not their spatial textual layout.
- 4) The lifetime of a global variable is the entire time of the program.
- 5) A strongly typed type is one that is either legal for the operator, or is allowed under language rules to be implicitly converted to a legal type.
- 6) Stack-dynamic variables are allocated storage when the block is entered and deallocated when it is exited.
- 7) A strongly typed language is one in which each name in a program has a single type associated with it and that type is known at compile time and all the type errors are detected prior to implementation / run time.
- 8) The explicit heap type variables are bound to storage that is allocated and deallocated by explicit run time instructions specified by the programmer.  
explicit heap type  $\rightarrow$  type error are always detected.

#### QUESTION TWO: TRUE / FALSE

- 9) T The main disadvantage of global variables is that they destroy the program modularity.
- 10) F A procedure P1 is a static parent of a procedure P2 if P2 calls P1.
- 11) F "\*" is bound to multiplication operation at compile time // language design time.
- 12) T Computing the average of 3 test grades: the expression "average=(grade1+grade2+grade3)/3.0" is more readable than "av =(g1+g2+g3) / 3.0".

Student Name:

Student id:

Section #:

University of Bahrain  
Department of Computer Science

College of Information Technology  
ITCS332: Concepts of Programming Languages

### QUIZZ#5: Chapter 5\_Names

QUESTION ONE: Fill in blanks

[6 pts]

- 7) A data type is compatible if it is either legal for the operator, or is allowed under language rules to be implicitly converted to a legal type.
- 8) The "INT" type in C++ is bound to a range of values at implementation time. The process of automatic conversion of a variable type to another type by a compiler is called coercion.
- 9) The type of an object can be determined using 3 ways: explicit (implicit) declaration statement, assignment statement, and operator statement.
- 10) A procedure P1 is a static parent of a procedure P2 if P1 contains the definition of P2. A procedure P1 is a dynamic parent of a procedure P2 if P1 calls P2.
- 11) The type of a variable defines: Range of values and Set of operators.
- 12) The lifetime of a static variable begins at compile time and ends at execution time.

### \* QUESTION TWO:

[4 pts]

Procedure main is

```
X : integer ;
Procedure sub1 is
Begin
  X = X - 8 ;
  Put (X) ;
End ;
Procedure sub2 is
  (X : integer ;
Begin
  X = 28 ;
  Sub1 ;
End ;
Begin
  X = 48 ;
  Sub2 ;
End ;
```

Consider the given Ada-like program. The value of X printed by put(X):

Under static-scoped rules is 20.

Under dynamic-scoped rules is 20.

Student Name:

Student id:

Section #: A

University of Bahrain  
Department of Computer Science

College of Information Technology  
ITCS332: Concepts of Programming Languages

### QUIZZ#5: Chapter 5\_Names

\*\*\*\*\*

#### QUESTION ONE:

[4 pts]

Procedure **main** is

X : integer;

Procedure **sub1** is

Begin

X = X \* 2;

Put (X);

End;

Procedure **sub2** is

X : integer;

Begin

X = 20;

Sub1;

End;

Begin

X = 50;

Sub2;

End;

Consider the given Ada-like program. The value of X printed by put(X):

Under static-scoped rules is 100 2

Under dynamic-scoped rules is 40 2

who  
calls me

9.5

Excellent

#### QUESTION TWO: Fill in blanks

[6 pts]

- 1) The main disadvantage of static variables is that they do not support recursion. The main disadvantage of global variables is that they destroy the program modularity.  
stack dynamic
- 2) The 2 kinds of type compatibility are: name compatibility and structure compatibility.
- 3) The explicit heap dynamic variables are bound to storage that is allocated and deallocated by explicit run time instructions specified by the programmer. stack dynamic variables are allocated storage when a program unit/ block is entered and deallocated when it is exited.
- 4) A call to a function is bound to the function code at link time. The symbol "+" is bound to the addition operation at language design time.
- 5) Dynamic Scope is based on the sequence of calls of program units, not their spatial textual layout.
- 6) In a dynamic scoping language, the referencing environment is the local variables plus all visible variables in all active subprograms.

Student Name:

Student id:

Section #: |

University of Bahrain  
Department of Computer Science

College of Information Technology  
ITCS332: Concepts of Programming Languages

### QUIZZ#6: Chapter 6\_Types

\*\*\*\*\*

- Name the 3 string length options used in programming languages:

1) Static length  
~~char~~ R = "SAT";

C++ example:

2) Dynamic length  
char R[5];

C++ example:

3) loss of Heap dynamic Memory  
char \*R;

C++ example:

- Given a matrix **U**: array  $\begin{matrix} & 20 & 60 \\ [0 \dots 19, 0 \dots 59] \end{matrix}$  of double; located at memory address starting at 2400. Element size is 5 bytes.

- a) Assuming column major ordering, calculate the address of matrix element U[12, 40].

$$2400 + ((11 \times 5) + 39 \times 5) = 2650$$

$$\text{Add}(i, j) = \text{Base} + [(j-1) \times n + (i-1)] \times \text{element size}$$

$$\text{Add}(12, 40) = 2400 + [39 \times 59 + 11] \times 5 = 13960$$

$$n = ? \\ 20 \text{ or } 59 \\ 59 \text{ or } 60$$

- b) Assuming row major ordering, calculate the address of matrix element U[12, 40].

$$\text{Add}(i, j) = \text{Base} + [(i-1) \times n + (j-1)] \times \text{element size}$$

$$\text{Add}(12, 40) = 2400 + [(11 \times 19) + (39)] \times 5 = 3640$$



Student Name:

Student id:

Section #:

University of Bahrain  
Department of Computer Science

College of Information Technology  
ITCS332: Concepts of Programming Languages

### QUIZZ#6: Chapter 6\_Types

\*\*\*\*\*

- Write C++ statements defining a fixed heap-dynamic array

`void Baller (id 50)`  
`{`

~~`int x = new [ ]`~~

`}`

~~`int x = new [ ]`~~

~~`int x = new [ 200 ]`~~

~~`int x = new [ 100 ]`~~; `int xx = new [ 100 ]`;

- Write C++ statements defining a static string length

`char str [ 8 ]`;

- Given a FORTRAN array definition: **DIMENSION Y (250)** and its base address is 8000, element size is 8 bytes, the address of the array element **Y(121)** is

$$= 8000 + 8 \times (121 - 1) = 9032$$

0          100          0          3          30



Given a matrix **U: array [120 .. 399, 120 .. 181] of FLOAT**; array's starting address is 2400; element size is 4 bytes. Show ALL your calculations.

- Assuming row major ordering, calculate the address of matrix element **U[250, 160]**.

$$[250, 160] = 2400 + 4 \times ((250 - 120) \times 2 + (160 - 120))$$

- Assuming column major ordering, calculate the address of matrix element **U[250, 160]**.

$$[250, 160] = 2400 + 4 \times ((160 - 120) \times 280 + (250 - 120))$$

$$2400 + 4 \times ((250 - 120) \times 280 + (160 - 120))$$



Student Name:

Student id:

Section #: 1

University of Bahrain  
Department of Computer Science

College of Information Technology,  
ITCS332: Concepts of Programming Languages

### QUIZZ#7: Chapter 7 Expressions & Assignments

\*\*\*\*\*

Consider the following code: `"int a, b; float c; c = a / b;"` and answer the next 2 questions (1 and 2).

- 1) If the assignment coercion is done before operand coercion, the value of c will be: float ~~int~~ ~~float~~
- 2) If the operand coercion is done before assignment coercion, the value of c will be: int ~~float~~
- 3) The use of an operator for more than one purpose is called operator overloading ~~operator overload~~
- 4) 2 disadvantages of operator overloading are: loss of compiler detection and loss of readability ~~error~~
- 5) Give a C++ expression that includes a binary operator: a = b - and another expression that includes a ternary operator: (flag) ? <sup>total=0</sup> exp1 : <sup>sum=0</sup> exp2
- 6) The order of operator evaluation in expressions depends on the Precedence and associativity rules.
- 7) The binary operator that appears between operands is called infix operator.  
The binary operator that precedes operands is called Prefix operator.
- 8) In C++, the type conversion which requires explicit code in programs is called Casting
- 9) A short circuit evaluation is one in which the result of an expression is determined without evaluating all of the operands and/or operators.
- 10) Functional side effects occur when a function changes a 2-way parameters or non-local variables.

Student Name:

Student id:

sect #:

serial #:

University of Bahrain  
Department of Computer Science

College of Information Technology  
ITCS332: Concepts of Programming Languages

**QUIZZ#4: Chapter 16 Prolog**

\*\*\*\*\*

**QUESTION:** Write Prolog rules (predicates) named double that takes a list of any numbers and produces a new list each element of which is equal to the double of the corresponding element in the original list.

?- double([3, 2.4, 1.25, 9] , U).  
U = [6,4.8,2.50,18]

**ANSWER:**

double([ ],[ ]).

double([A|B],[U|D]) :- U is A\*2, double(B,D).

Student Name:

Student id:

Section #:

University of Bahrain  
Department of Computer Science

College of Information Technology  
ITCS332: Concepts of Programming Languages

### QUIZZ#7: Chapter 7\_ASSIGNMENT STATEMENT & EXPRESSIONS

\*\*\*\*\*

Let the function **fun** be defined as

```
int fun (int *k)
{
    *k += 4; (14)
    return 3 * (*k) - 1 ;
}
void main()
{
    int i = 10, j = 10, sum1, sum2 ;
    sum1 = ( i / 2 ) + fun ( &i );
    sum2 = fun (&j ) + ( j / 2 );
}
```

What are the values of sum1 and sum2

- If the operands in the expressions are evaluated **left to right**?

$$i/2 = 5, \text{ fun}(\&i) = (3 \times 14) - 1 = 41$$

$$\text{sum1} = 5 + 41 = 46$$

sum2

$$\text{fun}(\&j) = 41, j = 14$$

$$\text{sum2} = 41 + (14/2) = 41 + 7 = 48$$

- If the operands in the expressions are evaluated **right to left**?

sum2

$$\text{sum2} = 41 + (14/2) = 41 + 7 = 48$$

$$\text{sum1} = 5 + 41 = 46$$

Student Name:

Student id:

sect #:

serial #:

University of Bahrain  
Department of Computer Science

College of Information Technology  
ITCS332: Concepts of Programming Languages

**QUIZZ#3: Chapter 16\_Prolog**

- \*\*\*\*\*
- 1) Given the list: `prophets = [ahmed, isa, [musa, hud, saleh]]`, the list head is ahmed, the list tail is [isa, [musa, hud, saleh]].
  - 2) Name 2 kinds of Prolog statements: Fact and rule.
  - 3) Prolog interpreter says no if the goal cannot be satisfied, and says yes if the goal can be satisfied.
  - 4) In a Prolog program, every fact or rule is terminated by .; to get the next answer for a given query, the user types ;.
  - 5) The process of finding a complete sequence of propositions (proof) for the first subgoal before working on others is called Depth First search; the process that works on all subgoals of a given goal in parallel is called Breadth First search.
  - 6) There are 2 approaches of matching a given goal to a fact in a database:  
bottom-up (forward) and Top-down (backward).
  - 7) The Prolog query: `"?- [99|T] = [H, u, 1, k, 3.2]."` produces:  
T = [u, 1, k, 3.2] yes H = 99 -0.5
  - 8) The two sides of a Prolog rule are separated by :- symbols. The terms in a disjunction are separated by ; symbols. -0.5
  - 9) The right side of rule is called antecedent, and the left side is called consequent.
  - 10) The name of a file containing a prolog program is written with an extension .pl.

Prolog

10

Student Name: \_\_\_\_\_

Student id: \_\_\_\_\_

Section #: 1

University of Bahrain  
Department of Computer Science

College of Information Technology  
ITCS332: Concepts of Programming Languages

**QUIZZ#3: Chapter 16 Prolog**

\*\*\*\*\*

a) Given the following Prolog clauses (predicates):

What2([ ], [ ]).

What2([A], [A,A]).

What2([A,B|C], [B,A|D]) :- what2(C,D).

What will be printed by the Prolog interpreter in each of the following 3 queries?

?- what2([1,a,2,b,3,c,4,d,5,f], LG).

LG = [a,1,b,2,c,3,d,4,f,5]  
Yes

?- what2([r,t,7,5], UG).

UG = [t,r,5,7]  
Yes

?- what2(A,B).

~~YES~~ B = [A,A]  
A = [A]  
Yes

or A = [ ]  
B = [ ]

(Predicates)

b) Define a Prolog predicate **interval(I,K,L)** that creates a list containing all integers within a given range. It takes two integers I and K such that  $I \geq K$  and returns a list L containing all consecutive integers from I down to K.

?- interval(10,5,LC).

LC = [10,9,8,7,6,5]

Yes

?- interval(81,76,YF).

YF = [81,80,79,78,77,76]

Yes

?- interval(2,9,Lst).

No

interval(I,K,L) :- L

~~interval(I,K,L) :-~~

~~I >= K.~~

~~I1 is I.~~

~~interval(I1,K,L).~~

~~I1 is I-1.~~

~~I1 is I-1.~~

interval(I,K,L) :-

I > K,

interval(I,K,[I|L]).

Student Name:

Student id:

Section #: |

University of Bahrain  
Department of Computer Science

College of Information Technology  
ITCS332: Concepts of Programming Languages

### QUIZZ#3: Chapter 16\_Prolog

\*\*\*\*\*

**QUESTION ONE:** Write a prolog program (Predicates) to obtain a list L2 by deleting a given element X from a given list L1. `del(X, L1, L2)`.

**QUESTION TWO:** Carefully study the following prolog program and find out what will be printed by prolog for each of the following queries.

What88(X,1,X).

What88(X,Y,Z) :- Y>1, Y1 is Y-1, What88(X,Y1,Z1), Z is X\*Z1.

?- What88(3,4,U).

U = 81.

yes.

?- What88(2.5,3,M).

M = 15.625

yes.

?- What88(2,2.5,N).

N = 8

yes.

$2^{2.5} = 2$

Student Name:

Student id:

Section #: A

University of Bahrain  
Department of Computer Science

College of Information Technology  
ITCS332: Concepts of Programming Languages

### QUIZZ#4: Chapter 16\_Prolog

\*\*\*\*\*

- 1) Given the list: `prophets = [ahmed, isa, musa, hud, saleh]`, the head is ahmed, the tail is [isa, musa, hud, saleh].
- 2) Name 2 types of Prolog statements: rules statement (headed) and fact statement (headed).
- 3) Prolog interpreter says no if the goal cannot be satisfied, and says yes if the goal can be satisfied.
- 4) In a Prolog program, every fact or rule is terminated by .;; To continue getting the next answer for a given query, the user types ;.
- 5) The process of finding a complete sequence of propositions (proof) for the first subgoal before working on others is called depth-first search; the process that works on all subgoals of a given goal in parallel is called breadth-first search.
- 6) There are 2 approaches of matching a given goal to a fact in a database:  
Backward chaining (top-down-resolution) and forward chaining (bottom-up resolution).
- 7) The Prolog query: `"?- [m|T] = [H, b, f]."` produces: H = m, T = [b, f] yes.
- 8) The two sides of a Prolog rule are separated by :-. The terms in a conjunction are separated by ,.
- 9) The right side of rule is called consequent, and the left side is called antecedence.
- 10) The name of a file containing a prolog program is written with an extension pl.